

# VisionApp

Ana Caballero Cano, *Universitat Autònoma de Barcelona*

**Resum** — VisionApp, és un projecte proposat per l'alumna Ana Caballero i consisteix en el desenvolupament d'un software d'accessibilitat per a dispositius *Android*, el qual està destinat per a persones amb diferents alteracions a la visió, com baixa visió, ceguera, daltonisme, presbícia (vista cansada), etc. L'objectiu del present projecte consisteix en el desenvolupament de diferents funcionalitats integrades en un única aplicació mòbil dirigida a un públic amb necessitats diverses, les quals es podrien ampliar en un futur, dotant el sistema de la possibilitat d'augmentar el text de forma considerable i amb indicacions auditives.

Per una banda, s'ha desenvolupat el detector de llum que permet saber a l'usuari la quantitat de llum que hi ha en un lloc determinat. Per altre banda, el detector de color, informa a l'usuari dels colors captats per la càmera del dispositiu. A més, lupa facilita la lectura de documents augmentant el contrast amb els filtres de colors. Aquests colors de filtre són els mateixos que els utilitzats a les tel·lupes existents al mercat per a baixa visió. Finalment, l'usuari podrà configurar la interfície mitjançant un seguit de paràmetres predeterminats com la mida de la lletra o el mode del flash (en cas que el dispositiu en tingui).

**Paraules clau** — accessibilitat, alteracions a la visió, funcionalitats integrades, detector de llum, detector de color, lupa, filtres de colors.

**Abstract** — VisionApp, is a project proposed by the student Anna Caballero and involves the development of accessibility software for Android devices, which is designed for people with different vision disorders, such as low vision, blindness, color blindness, presbyopia (eye strain), etc. The aim of this project is the development of different functionalities integrated into a single mobile application aimed to an audience with diverse needs, which could be extended in the future, giving the system the ability to significantly increase the text and hearing directions.

On one hand, a light detector has been developed that allows the user to know the amount of light that is in a certain place. On the other hand, the color detector tells the user of the colors captured by the camera device. In addition, the lens provides reading documents easier by increasing the contrast with color filters. These colors filters are the same as those used in *teloupes* available on the market for poor vision. Finally, the user can configure the interface through a series of default parameters such as font size or flash mode (if the device has one).

Key words— accessibility, changes in vision, integrated functionalities, light detector, detector color, magnifying glass, color filter



## 1 INTRODUCCIÓ

ACCEDIR a tecnologies modernes, com *smartphones* o *tablets*, per part de persones d'edats i necessitats diferents ja no representa un problema d'excés de complexitat gracies a grans empreses com *Google Inc.* o *Apple* que han invertit esforços en el desenvolupament d'aplicacions i serveis d'accessibilitat orientat a persones amb necessitats especials [1].

Si fem la vista enrere, ha passat menys d'un segle, des del 1923. Any que el cos de policia de Victoria (Austràlia) utilitzà els primers telèfons mòbils [2], fins a dia d'avui. En els darrers anys, els telèfons mòbils s'han anat transformat en petits ordinadors (*smartphones*) augmentant exponencialment els seus possibles usos. Esdevenint, alhora, un element més en la vida quotidiana d'algunes persones.

Ja sigui per moda o necessitat, és cert que aquests invents poden facilitar la vida a les persones d'una amplitud

de formes molt diferents. No obstant, el món està fet a mida d'una majoria de persones "*estàndard*", o en altres paraules, persones sense limitacions. Per aquest motiu, és de suma importància posar els medis pertinents per tal de garantir l'accés a aquesta secció minoritària de la població a tecnologies modernes de mode que els hi permeti guanyar també en qualitat de vida d'una forma innovadora, tal que abans de l'aparició d'aquests dispositius moderns no era possible.

En aquest sector minoritari, es troben persones amb discapacitats visuals, físiques o relacionades amb l'edat.

Per aquestes persones, L'ús de tecnologies haurà de tractar-se d'una cosa trivial, no una gesta, igual que per aquells sense dificultats afegides.



Fig. 1. Aplicacions d'accessibilitat. D'esquerra a dreta: VizWiz (respon preguntes sobre l'entorn), Big Launcher (canvia l'aspecte de la interfície a una més senzilla), ScanLife Barcode (proporciona informació sobre el producte a que pertany el codi de barres o QR), NoLED (proporciona notificacions visuals, mitjançant LED o icones).

Aquest article pretén explicar, d'una banda, què ha motivat la realització d'aquest projecte i, per un altre banda, en què ha consistit el treball realitzat.

Començaré fent un breu estudi de l'estat de l'art i exposant els objectius que es van fixar per al projecte. A continuació, explicaré la metodologia utilitzada, la planificació seguida i les eines utilitzades per assolir els objectius proposats. Finalment, presentaré els resultats obtinguts i tancaré l'article amb una conclusió del treball.

## 2 ESTAT DE L'ART

A dia d'avui, els usuaris de dispositius mòbils amb requeriments especials disposen d'una ampla oferta d'aplicacions gratuïtes i de pagament enfocada a cobrir necessitats reals [3].

No obstant, aquestes aplicacions estan principalment dirigides només a una funcionalitat, el que implica que si l'usuari té diverses necessitats ha de descarregar múltiples aplicacions omplint ràpidament la memòria del dispositiu, alhora que augmenta la complexitat d'ús, ja que cadascuna d'aquestes aplicacions disposa d'una interfície diferent. Aquesta característica perjudica especialment a persones grans o amb problemes greus de visió degut a que els hi suposa un esforç més gran fer ús d'aquests dispositius. Per tant, en aquests cassos, resulta necessari comptar amb una aplicació que englobi diverses funcionalitats, donant solucions a diverses alteracions a la vista.

Amb el sistema *VisionApp*, es pretén proporcionar als usuaris de dispositius *Android* d'una eina que cobreixi múltiples deficiències visuals (figura 2):

- Detector de llum, per usuaris cecs o molt baixa visió, que els permetrà saber si el lloc on es troba està il·luminat, estalviant en electricitat.
- Detecto de color, per als usuaris esmentat al punt anterior i/o daltònics, dotant-los d'una major autonomia a l'hora de comprar roba, per exemple, el qual pot esdevenir un veritable mal de cap.
- Lupa, permet a usuaris vidents però amb dificultats augmentar la mida i el contrast d'un text per

facilitar la seva lectura, tal com ho fa una tel·lupa [4] però de forma més lleugera.

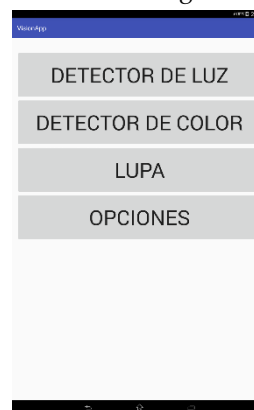


Fig. 2. Menú principal de *VisionApp*, on es mostren les diferents funcionalitats integrades.

## 3 OBJECTIUS

L'objectiu principal d'aquest projecte era la creació de l'aplicació *VisionApp*, que integra diversos mòduls. Aquest mòduls, estan dirigits a persones amb afectacions visuals de diversos índoles. Inicialment, el sistema s'ha implementat per a persones amb ceguera, baixa visió i daltonisme però es podria ampliar en un futur i expandir l'aplicació a un mercat és ampli. Per aquest motiu es van definir els següents objectius, més específics, pel que fa als requisits del sistema:

- Determinar la quantitat de llum que hi ha per saber si el llum de la sala està engegat o aturat.
- Realitzar un zoom de la imatge per tal d'enforçar una zona determinada de l'escena.
- Activar el flash del dispositiu de forma manual i automàtica en funció de la llum ambient.
- Capturar la imatge clicant a qualsevol posició de la pantalla per tal de simplificar el seu ús per a persones amb baixa visió.
- Segmentar els colors de la imatge i comunicar l'usuari els colors trobats.
- Aplicar filtres a la imatge per augmentar el contrast i facilitar la lectura de documents.

## 4 METODOLOGIA

Per tal d'aconseguir els objectius marcats, es seguirà una versió adaptada de la metodologia SCRUM [5].

Ja que aquest projecte està integrat dins del desenvolupament de l'assignatura universitària *Treball Final de Grau* (TFG) [6] els rols de *Development Team* (equip de desenvolupament) i *Scrum Master* estan representats per l'alumna Ana Caballero. Al tractar-se d'un projecte proposat per part de l'alumna el rol de *Product Owner* (propietari del producte) estarà compartit per l'alumne abans esmentat i pels tutors del projecte Jordi Roig i Marc Vallribera.

Els cicles de treball s'han organitzat en *sprints* o iteracions, determinats principalment per les fites de l'assignatura 'Treball de Fi de Grau'. Durant cada *sprint*, es duen a terme unes tasques específiques, *sprint backlog*, per assolir els objectius establerts prèviament, *product backlog*, generant un increment de software potencialment lliurable. Al principi i final de cada *sprint* es duen a terme reunions amb els diferents rols implicats per a revisar el treball realitzat fins a la data, es plantejaven els dubtes sorgits i es planificava el treball a realitzar a continuació, procurant seguir la planificació establerta. Els tutors, per la seva banda, s'encarregaven de valorar la feina i d'assenyalar les possibles millores o canvis a aplicar. De forma paral·lela i menys formal, hem anat mantenint una breu reunió setmanal que ha estat especialment important a l'inici del desenvolupament del projecte per tal d'evitar problemes futurs.

La figura 3 mostra de forma esquemàtica el procés de desenvolupament de SCRUM.

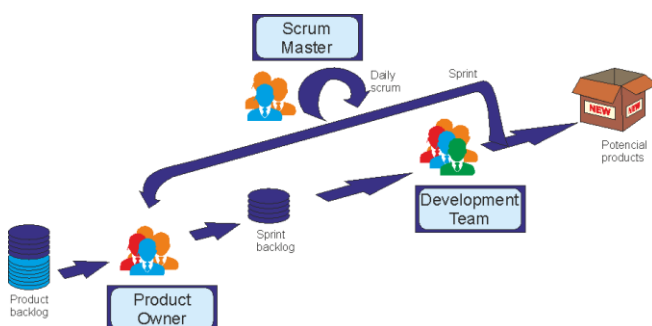


Fig. 3. En la metodologia de desenvolupament SCRUM, un projecte s'executa en blocs temporals curts i fixes (1 a 4 setmanes), on cada iteració ha de proporcionar un increment de producte final que sigui susceptible de ser modificat o lliurat amb el mínim esforç al client, quan aquest ho sol·liciti. Aquest procés parteix de la llista d'objectius i prioritats del producte, que actua com a pla de projecte.

Aquesta metodologia de desenvolupament ha estat de gran utilitat, ja que s'ha pogut obtenir un feedback ràpid per part dels tutors, la qual cosa va permetre una gestió regular de les expectatives del projecte, minimitzant possibles riscos. Com per exemple substituir alguna funcionalitat per l'accessibilitat, necessària però no contemplada en la planificació del primer informe.

## 5 PLANIFICACIÓ I EVOLUCIÓ

Aquest projecte s'ha desenvolupat des de setembre de 2015 fins a gener de 2016, amb una dedicació aproximada de 20 hores setmanals i 400 hores totals. El treball està dividit en 4 sprints, que detallats a continuació.

A la figura 4 podem observar el diagrama de Gantt que representa la planificació temporal del projecte.

			Set	Oct	Nov	Dic	Ene
	Inici	Final					
TFG	14/09/15	29/01/16					
Sprint 0	14/09/15	27/09/15					
Sprint 1	19/10/15	06/11/15					
Sprint 2	19/11/15	04/12/15					
Sprint 3	07/12/15	29/01/16					

Fig. 4. Planificació temporal del projecte estructurada en *sprints*.

En l'*sprint* 0, s'investiga l'estat de l'art i es varen especificar els requisits preliminars del sistema, motiu pel que es van valorar diferents entorns de desenvolupament multiplataforma [7], ja que inicialment el sistema estava enfocat per a poder-se utilitzar en diversos sistemes operatius. Amb l'objectiu d'establir una proposta detallada del projecte, es va elaborar un *Informe Inicial* i deixant constància de la reunió en una acta.

Durant l'*sprint* 1 es va procedir a implementar la funcionalitat de la lupa amb els diferents filtres de colors amb HTML5 [8], CSS [9] i JavaScript [10] ja que aquest llenguatge són interpretats per tots els dispositius de forma equànime. Però a l'hora d'aplicar els filtres es requeria accedir a la càmera del dispositiu [11], el qual només es pot dur a terme en codi natiu. Aquest fet va afectar a tota la planificació inicial del projecte, havent de tornar enrere i forçant a triar un sistema operatiu concret per desenvolupar el projecte. Finalment es va optar per *Android* natiu i s'implementà la lupa amb els filtres propis d'una tel·lupa i el sensor de llum. A més, es va redactar l'*Informe de Progrés I*, en el qual es va deixar constància dels avenços efectuats en el desenvolupament del projecte.

Al llarg del *sprint* 2 es va procedir, en primer lloc, a millorar el temps de processament de la imatge en el moment d'aplicar els filtres ja que el temps invertit en el procés no era òptim. Aquesta millora es va produir gràcies a la reducció de les dimensions de la imatge a una resolució VGA [12] (64 x480). Es van provar altres mètodes com l'ús de *onPreviewFrame*, però no es crida mai amb *setPreviewCallback*, degut a un problema de compatibilitat amb *Android* 4 [13] (la versió més extensa actualment). També es fa dur a terme una quantització [14] de la imatge capturada abans de processar-la per tal de reduir la mida de la matriu que conté la imatge, mostrejant-la, però la millora en temps no era significativa i la pèrdua d'informació (qualitat) era important. Per aquest motiu també es va descartar. Un cop optimitzat el temps d'aplicar filtres de colors, es va procedir a fer el disseny final de l'aspecte de la interfície. Finalment, es va tancar aquesta fase amb l'elaboració de l'*Informe de Progrés II*, en què s'especificava el treball efectuat en aquesta etapa i s'exposaven els resultats i conclusions preliminars.

El *sprint* 3 es va dedicar a la realització del menú d'opcions i la comunicació de missatges per veu, gràcies al *Text to Speech (TTS)* [15]. També, es va implementar el detector de colors i el manual d'usuari. Finalment, aquest article es va redactar en conculre aquesta fase. Les darreres tres setmanes s'han reservat per elaborar el dossier del TFG, una síntesi de tot el treball realitzat, i per preparar la defensa

pública del projecte en una presentació davant d'un comitè. avaluador.

No s'ha pogut seguir la planificació prevista a l'inici del projecte, concretada a l'*Informe de Progrès I*, ja que aquesta era massa ambiciosa i no contemplava possibles handicaps, dissenyar la interfície i fer l'aplicació accessible. Possibles aplicacions que s'havien de dur a terme durant el *sprint 2* han estat de substituïts per d'altre com optimitzar el temps de processament, no contemplades a la planificació inicial. Els obstacles que han portat a la modificació de la planificació són deguts a que l'estudiant no estava familiaritzat amb el llenguatge de programació i requeria una continua investigació per trobar les eines més òptimes possibles.

## 6 EINES UTILITZADES

Per a la realització del present projecte s'han utilitzat les eines que es comenten a sota.

### 6.1 Entorns de programació

#### 6.1.1 PhoneGap / Apache Cordova

Es tracten d'una mateixa plataforma, tot i tenir noms i poder-se descarregar de llocs diferents. Originalment, *Nitobi* creà *PhoneGap* que va ser absorbit per *Adobe* el 2012 i reanomena el projecte com a *Cordova* amb finalitats comercials [16].



Fig. 5. Entorn de programació multiplataforma *PhoneGap/Cordova*.

Actualment, *Cordova* es tracta d'un entorn de desenvolupament lliure que permet crear aplicacions mòbils utilitzant APIs de web estandarditzats per a projectes propis.

No es programa directament en *PhoneGap*. Aquest utilitza codis desenvolupats, sense la necessitat de *Software Development Kits (SDKs)* natius, en els llenguatges *HTML5*, *CSS* i *JavaScript* i els compila en el núvol per les plataformes més comunes del mercat: *Android*, *iOS* i *Windows Phone 8* (les plataformes *Blackberry*, *Symbian*, *WebOS* van quedar descatalogades a partir de la versió 3.0) [17].

Es va descartar per que no permetia accedir per codi al hardware de la càmera del dispositiu.

#### 6.1.2 Xamarin

En la mateixa direcció que *Cordova*, *Xamarin* [18] és un entorn de desenvolupament integrat (IDE) que fa servir *C#* en comptes d'utilitzar *HTML5* per generar aplicacions compatibles en les mateixes tres plataformes natives.

També es va descartar per problemes amb l'emulador, a més que tampoc permetia accedir a la càmera del dispositiu.



Fig. 6. Entorn de programació multiplataforma *Xamarin*.

#### 6.1.3 Eclipse

Es tracta d'un *Entorn de Desenvolupament Integrat (IDE)* de codi obert multiplataforma. És un dels entorns més utilitzats per *Java*, *C/C++*, *JavaScript* i *PHP*. El paquet bàsic es pot personalitzar a mida de les necessitats de l'usuari instal·lant *plugins* que permeten afegir funcionalitats. Per aquest projecte, per *Android*, va ser necessari el plugin ADT de *Android*.



Fig. 7. Entorn de programació multiplataforma *Eclipse*.

Per treballar amb *PhoneGap*, es recomanava triar entre *netBeans* i *Eclipse*. Es va optar per aquest per que a la alumna ja havia treballat prèviament amb *Eclipse* i tenia un major domini del programa. Al fer la migració d'una aplicació multiplataforma a una només per *Android*, va quedar descartat el seu ús.

#### 6.1.4 Notepad++

*Notepad++* és un editor de text i codi font optimitzat per a *Windows* que utilitza per una ampla comunitat de programadors, com a substitut del *Notepad*.

Es va utilitzar abans de descarregar *Eclipse* per a la programació en *HTML5*, però es va tornar a utilitzar més tard de forma paral·lela amb *Android Studio* amb el plugin *TextFX* per fer un *reident* (tabulació correcta) del codi.

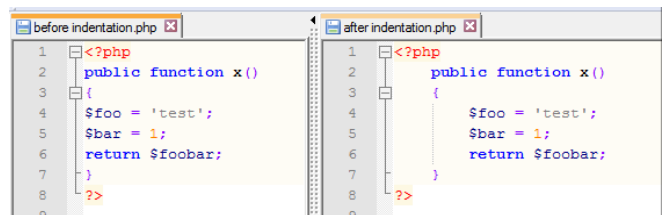


Fig. 8. Exemple d'un fitxer en *Notepad++* abans i després d'aplicar *reident* mitjançant el plugin *TextFX*.

#### 6.1.5 Android Studio

*Android Studio* és l'IDE oficial per al desenvolupament d'aplicacions per *Android* i ha estat l'entorn de programació principal utilitzat al llarg del projecte.

Utilitza plantilles de codi tant per la mida del dispositiu al qual està adreçat (telèfons *Android*, *tablets*, *Android TV*, *Android Auto* i *Google Glass*).

*Android Studio*, permet la importació d'exemples de codi de *Google* des de *GitHub* i generar l'aplicació (fitxer *APK*).



Amb la finalitat d'agilitzar el procés de simulació per comprovar el correcte funcionament del sistema es va fer servir una tablet *Sony Expertia Z* dotat amb *Android 4.4*, en comptes de l'emulador integrat al programa. Les aplicacions desenvolupades en *Android Studio* es desenvolupen en *Java*.



Fig. 9. A la esquerra: Entorn de desenvolupament oficial d'Android, *Android Studio*. A la dreta: Model del dispositiu utilitzat per fer proves al llarg del projecte.

## 6.2 Llenguatges de programació

### 6.2.1 HTML5

*HTML5* és el llenguatge de programació web per excel·lència. Proveeix bàsicament tres característiques: estructura, estil i funcionalitat. *HTML5* és el resultat de la combinació d'*HTML*, *CSS* i *JavaScript*. Aquestes tecnologies són altament dependents i actuen com una sola unitat organitzada sota la especificació *HTML5*.



Fig. 10. *HTML* està a càrrec de l'estructura, *CSS* de la presentació mostrada per pantalla i *JavaScript* dota de dinamisme el lloc web

Aquest llenguatge va ser utilitzat durant la primera versió multiplataforma, però es va haver d'abandonar per poder accedir al hardware del dispositiu ja que aquest llenguatge no ho permet.

### 6.2.2. Java

*Java* [19] és un llenguatge de programació orientat a objectes creat amb la finalitat reduir dependències d'implementació. La seva finalitat, és permetre que els desenvolupadors d'aplicacions escriguin el programa un sol cop i aquest es pugui executar des de qualsevol dispositiu sense la necessitat de compilar-lo per a cada plataforma diferent.



Fig. 11. Controlador *Java* per *Android*.

El sistema *VisionApp* s'ha desenvolupat en *Java*, ja que les aplicacions per a *Android* se desenvolupen, principalment,

en aquest llenguatge.

## 6.3 Altres

### 6.3.1. OpenCV

Es tracta d'una llibreria de programació lliure multiplataforma (*C/C++*, *Python* i *Java*), enforcada principalment en el processament d'imatges en temps real.

*OpenCV* [20] és la biblioteca de codis més popular i avançada per a aplicacions relacionades amb la Visió per Computador avui dia, que dona suport a tasques molt bàsiques (captura i processament de dades de la imatge) com a algorismes d'alt nivell (extracció de característiques, seguiment de moviment, aprenentatge automàtic, etc.).

Aquesta llibreria ha estat imprescindible a l'hora processar les imatges i aplicar filtres de colors a la càmera, en l'opció de lupa.



Fig. 12. Transformacions necessàries per convertir la imatge obtinguda per la càmera a la imatge amb filtres de colors.

### 6.3.2 Corel Draw

*Corel Draw* és una eina de disseny gràfic, permet realitzar il·lustracions lineals i edició professional de fotografies.

S'ha recorregut a aquesta eina, ja que l'alumna te coneixements previs del seu funcionament i permet generar fitxers d'extensió *.png* necessaris per a generar els icones de l'aplicació.



Fig. 13. A la esquerra: Eina de disseny *Corel Draw*. A la dreta: Icona *VisionApp*. Els colors estan triat per a que una persona amb daltonisme els pugui percebre.

## 7. RESULTATS

A continuació es mostren els resultats obtinguts del sistema desenvolupat.

### 7.1. Detector de llum

A la següent simulació, l'usuari amb baixa visió o invident vol saber el lloc on es troba està il·luminat de forma intensa, en penombra o es troba a les fosques. Per a saber-ho accedirà a la primera opció del menú principal.

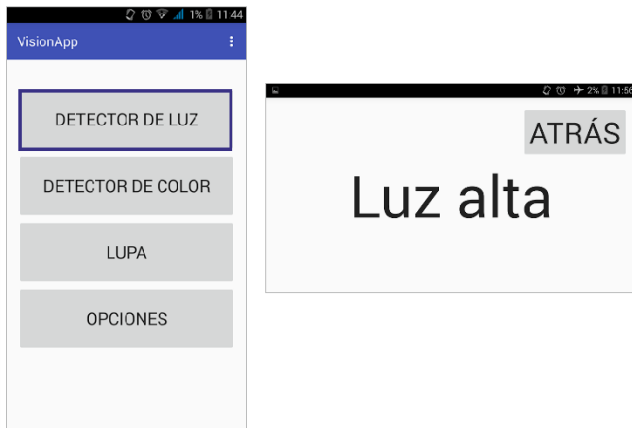


Fig. 14. A la esquerra: Menú principal amb l'opció *Detector de llum* marcada. A la dreta: Possible sortida retornada pel sistema.

L'aplicació comunicarà a l'usuari tres possibles valors, per text i veu, en funció de la intensitat de la llum ambient. Aquest valors són: Llum nul·la (fins a 3%), llum baixa (entre 4% i 10%) i llum alta (a partir del 11).

## 7.2. Detector de color

En aquest altre exemple, l'usuari desitja saber els colors d'un objecte captat amb la càmera del seu dispositiu, ja sigui per que no hi veu o no percep un color (daltonisme). Per a descobrir-ho haurà d'accedir a la segona opció del menú principal.

Un cop obert seleccionat el detector de color, l'usuari podrà conèixer els colors de la imatge, només haurà de polsar-la

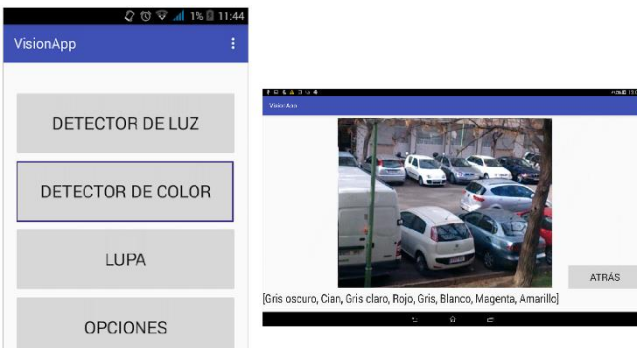


Fig. 15. A la esquerra: Menú principal amb l'opció *Detector de color* marcada. A la dreta: Sortida retornada pel sistema per a la imatge captada.

L'aplicació comunicarà a l'usuari, per text i veu, els colors trobats per la càmera en el moment de seleccionar el botó *Colors*. Els colors en format text apareixeran sota els botons.

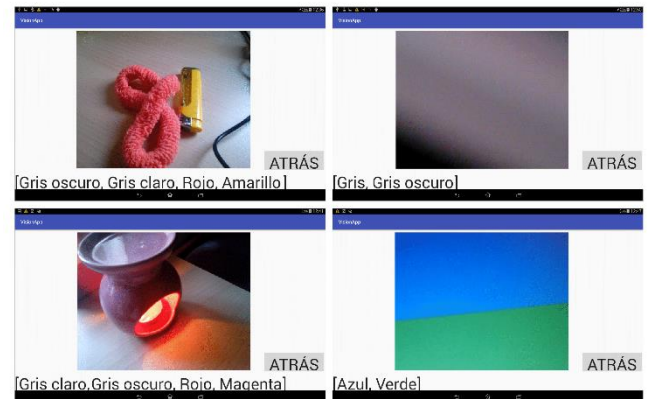


Fig. 16. Exemples d'imatges capturades i les sortides per text corresponents.

## 7.3. Lupa amb filtres

Quan l'usuari de l'aplicació desitgi llegir un text però li resulti difícil, ja sigui per baixa visió o poc contrast entre el fons i el text, pot utilitzar la opció *Lupa*, tercera opció del menú principal.

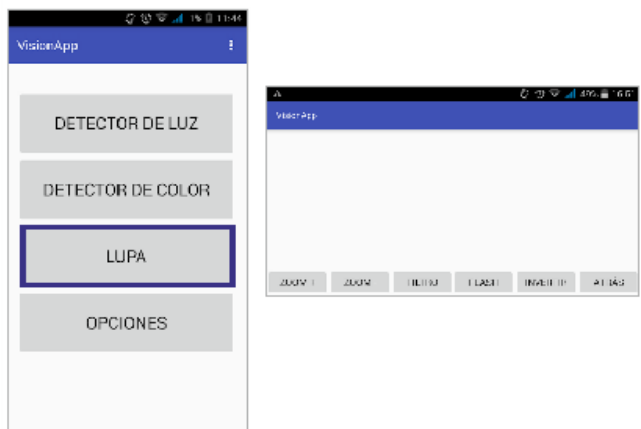


Fig. 17. A la esquerra: Menú principal amb l'opció *Lupa* marcada. A la dreta: Lupa abans d'activar-se la càmera amb filtres.

Un cop obert activada la càmera, polsant enmig de la pantalla, l'usuari podrà acostar o allunyar la imatge, modificar el mode del flash i invertir els colors de la imatge. Per canviar el filtre aplicat, ho podrà fer polsant el botó *Filtre*.

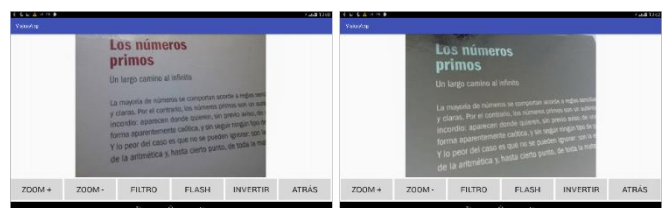


Fig. 18. A la esquerra: Exemple de la contraportada d'un llibre capturat sense filtres. A la dreta: Mateixa contraportada sense filtres i colors invertits.



Fig. 19. Exemple anterior amb els diferents filtres i els seus inversos corresponents (a la dreta), en ordre d'aparició al sistema. De dal a baix: blanc, verd, taronja, cian (blau clar), magenta (rosa), blau i groc.

Els filtres anteriors, estan basats en sistemes existents. Aquests sistemes, consisteixen en diferents tipus de telelupes a la venda, actualment, per internet i una altre telelupa de la que disposa el tutor del projecte, Jordi Roig.

Aquests color estan comprovats que són els que una persona amb problemes visuals millor distingeixen.

Per exemple, una persona a la que li enlluerni molt la llum del monitor, un filtre adient seria el blau amb negre, però una altra que requereix un major contrast potser aplicaria el verd amb negre. Si a més de baixa visió, es tracta d'una persona amb daltonisme pels verds o vermells, els filtres verd i magenta no serien convenients, però podria aplicar un filtre groc més viu i que és un que el pot identificar correctament..

## 7.4. Opcions

L'usuari pot configurar l'aplicació amb un seguit de valors predeterminats, que es mantindran de forma constant cada cop s'obri la aplicació de nou.

Podrà triar l'idioma (alemany, espanyol, francès, anglès o italià), la mida del text (normal, gran o molt gran), el mode del flash (si el dispositiu en disposa, automàtic, encès o aturat) i el color de filtre per defecte (sense filtre, blanc, verd, taronja, cian, magenta, blau o groc).

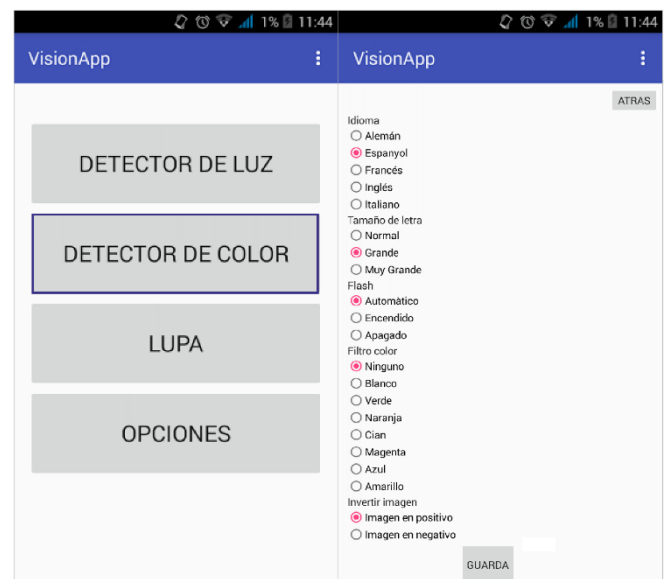


Fig. 20. A la esquerra: Menú principal amb l'opció *Opciones* marcada. A la dreta: Menú d'opcions.

## 7.5. Manual

Al manual d'usuari es pot accedir des del menú principal o des de opcions, fent tocant al *menú Overflow* (tres punts de la barra superior). El manual conté informació útil del funcionament del sistema.

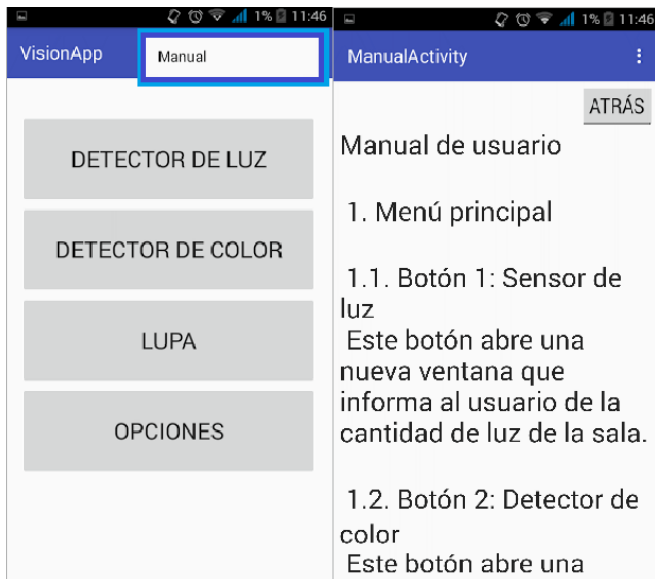


Fig. 21. A la esquerra: Menú principal amb l'opció *Manual* visible i marcada. A la dreta: Manual d'usuari.

## 8 CONCLUSIONS

Després del *Informe Inicial*, el projecte es va veure retallat en funcionalitats degut als canvis en la planificació (veure 5. *Planificació i evolució*). A més, s'ha inclòs accessibilitat i un menú d'opcions no contemplat a la planificació inicial.

A més, s'han aconseguit els objectius crítics proposats a la fase inicial, tret del OCR [21] que ha quedat descartat, de moment, per qüestions de planificació. Per tant, el sistema VisionApp es troba totalment funcional.

Sens dubte, es pot afirmar que VisionApp tindrà un impacte positiu en l'usuari, ja que disposarà d'una única aplicació que engloba diverses funcionalitats, donant solucions a diverses alteracions a la vista, i estalviant espai de memòria en el dispositiu. Alhora, al integrar diverses funcions en un mateix sistema es simplificarà el seu ús, per que totes les funcionalitats segueixen una mateixa estructura.

Pel que fa a les possibles extensions, el projecte es tindria dues possibles ampliacions. Per una banda, la implementació d'un OCR amb sortida per veu, plantejat inicialment però deixat en segon pla per manca de temps a la planificació. Per altre banda, una càmera que altera els canals RGB de la imatge, tal com ho fa *EnChroma* [22]. *EnChroma* va llançar a finals del primer trimestre del any 2015 unes ulleres que modifiquen la l'espectre de llum visible i permeten a persones amb daltonisme percebre la imatge amb els colors igual que una persona que pot percebre tots els colors

## 9. AGRAÏMENTS

Voldria expressar el meu reconeixement a totes aquelles persones que, d'una manera o altra, han contribuït a dur a terme el present *Treball Final de Grau*.

En primer lloc, el meu més sincer agraïment a Jordi Roig i Marc Vallribera, tutors d'aquest projecte, per tenir sempre la porta oberta als meus dubtes i pels seus valuosos consells al llarg d'aquesta etapa.

També voldria agrair a la meua família i amics pel seu suport i ànims brindats durant la carrera igual que al llarg d'aquest projecte.

A tots ells els hi vull donar les gràcies.

## REFERÈNCIES

- [1] DiscapNet. APPS Accesibles - Utilidades. [http://www.discapnet.es/Castellano/areastematicas/tecnologia/aplicaciones\\_moviles/Paginas/APPS-Accessibles.aspx](http://www.discapnet.es/Castellano/areastematicas/tecnologia/aplicaciones_moviles/Paginas/APPS-Accessibles.aspx) [Online. Accedit 18/01/2016].
- [2] Victoria Police. Victoria Police History. [http://www.police.vic.gov.au/content.asp?Document\\_ID=12325](http://www.police.vic.gov.au/content.asp?Document_ID=12325) [Online. Accedit 18/01/2016].
- [3] Omicromo. Soluciones para sordos, ciegos, mayores y gente con movilidad reducida. <http://www.omicrono.com/2013/01/smartphones-y-accesibilidad-soluciones-para-sordos-ciegos-mayores-y-gente-con-movilidad-reducida/> [Online. Accedit 18/01/2016].
- [4] Rehasoft - Accesibilidad. Telelupas. <http://www.rehasoft.com/baja-vision/telelupas/> [Online. Accedit 18/01/2016].
- [5] IEEE Software (Juliol/Agost 2000). El procés de desenvolupament de software Scrum per a petits equips. <http://web.lindarising.info/uploads/IEEEScrum.pdf> [Online. Accedit 18/01/2016].
- [6] Universitat Autònoma de Barcelona. Treball Final de Grau (2015). <http://www.uab.cat/web/estudis/treball-de-fi-de-grau1345688899319.html> [Online. Accedit 18/01/2016].
- [7] eSandra. 10 Soluciones para Crear Apps Multiplataforma. <http://www.esandra.com/soluciones-para-crear-aplicaciones-movilesmultiplataforma> [Online. Accedit 18/01/2016].
- [8] W3Schools. HTML(5) Tutorial. <http://www.w3schools.com/html/default.asp> [Online. Accedit 18/01/2016].
- [9] W3Schools. CSSTutorial. <http://www.w3schools.com/css/default.asp> [Online. Accedit 18/01/2016].



- [10] W3Schools. JavaScript Tutorial.  
<http://www.w3schools.com/js/default.asp>  
[Online. Accedit 18/01/2016].
- [11] Iteramos. Cuáles son las limitaciones y desventajas de Phonegap o HTML5 en comparacion a los de Android.  
<http://www.iteramos.com/pregunta/41573/cuales-son-las-limitaciones-y-desventajas-de-phonegap-o-html5-en-comparacion-a-los-de-android>  
[Online. Accedit 18/01/2016].
- [12] IBM VGA Technical Reference Manual.  
[http://www.mcarnafia.de/pdf/ibm\\_vgaxga\\_trm2.pdf](http://www.mcarnafia.de/pdf/ibm_vgaxga_trm2.pdf)  
[Online. Accedit 18/01/2016].
- [13] Android Open Source Project - Issue Tracker.  
<https://code.google.com/p/android/issues/detail?id=20999#makechanges>  
[Online. Accedit 18/01/2016].
- [14] Universitat d'Oviedo - Lección.3: La Transformada. Efectos del muestreo y la cuantización.  
<http://www6.uniovi.es/vision/intro/node15.html>  
[Online. Accedit 18/01/2016].
- [15] Text-To-Speech.  
<https://sites.google.com/site/herramientasidiomas/4a-sesion>  
[Online. Accedit 18/01/2016].
- [16] Adobe Announces Agreement to Acquire Nitobi, Creator of PhoneGap  
<http://www.adobe.com/aboutadobe/pressroom/press-releases/201110/AdobeAcquiresNitobi.html>  
[Online. Accedit 18/01/2016].
- [17] PhoneGap Build. Blackberry, Symbian, and WebOS will not be supported by PhoneGap 3.0 .  
[http://community.phonegap.com/nitobi/topics/blackberry\\_symbian\\_and\\_webos\\_will\\_not\\_be\\_supported\\_by\\_phonegap\\_3\\_0](http://community.phonegap.com/nitobi/topics/blackberry_symbian_and_webos_will_not_be_supported_by_phonegap_3_0)  
[Online. Accedit 18/01/2016].
- [18] Xamarin. Create native iOS, Android, Mac and Windows apps in C#.  
<https://xamarin.com/platform>  
[Online. Accedit 18/01/2016].
- [19] Oracle Corporation. Java™ Platform, Standard Edition 8 API Specification.  
<https://docs.oracle.com/javase/8/docs/api/index.html>  
[Online. Accedit 18/01/2016].
- [20] OpenCV. Tutorial: OpenCV4Android SDK.  
[http://docs.opencv.org/2.4/doc/tutorials/introduction/android\\_binary\\_package/O4A\\_SDK.html](http://docs.opencv.org/2.4/doc/tutorials/introduction/android_binary_package/O4A_SDK.html)  
[Online. Accedit 18/01/2016].
- [21] ABBYY. Que es Reconocimiento óptico de caracteres (OCR).  
<http://es.abbyy.com/finereader/ocr/>  
[Online. Accedit 18/01/2016].
- [22] EnChroma. Color for the color Blind™.  
<http://enchroma.com/>  
[Online. Accedit 18/01/2016].